PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 5 : G06F 11/34 — A1

(11) International Publication Number: WO 95/00906

(43) International Publication Date: 5 January 1995 (05.01.95)

(21) International Application Number: PCT/US94/07009

(22) International Filing Date: 21 June 1994 (21.06.94)

(30) Priority Data:
08/081,391 23 June 1993 (23.06.93) US

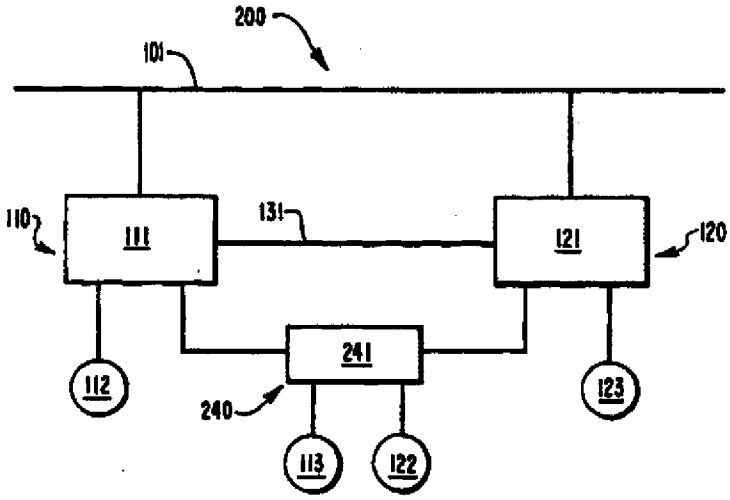
(71) Applicant: VINCA CORPORATION [US/US]; 4000 Central Park East, 1815 South State Street, Orem, UT 84058 (US).

(74) Agents: CHRISTIANSEN, Jon, C. et al.; Van Cott, Bagley, Cornwall & McCarthy, 50 South Main Street, Suite 1600, Salt Lake City, UT 84144 (US).

(81) Designated States: AT, AU, BB, BG, BR, BY, CA, CH, CN, CZ, DE, DK, ES, FI, GB, HU, JP, KP, KR, KZ, LK, LU, LV, MG, MN, MW, NL, NO, NZ, PL, PT, RO, RU, SD, SE, SK, UA, US, UZ, VN, European patent (AT, BE, CH, DE, DK, ES, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, ML, MR, NE, SN, TD, TG).

Published
With international search report.

(54) Title: METHOD FOR IMPROVING DISK MIRRORING ERROR RECOVERY IN A COMPUTER SYSTEM INCLUDING AN ALTERNATE COMMUNICATION PATH



(57) Abstract

A method for reducing the time necessary to recover from a processor (111, 121) failure in a fault-tolerant computer system with redundant server computer systems (110, 120) with their own disk storage systems is disclosed and claimed. In normal operation whenever data is to be written to disk storage, each of the servers writes an identical copy of the data to its own disk storage system. When a server processor fails and then is restored to operation, that server's disk storage system must be made identical to (consistent with) the disk storage system of the non-failing server before the system is again fault tolerant. This method improves performance by electronically transferring the disk storage system from the failing server to a non-failing server, having the non-failing server keep the transferred disk storage system identical to its normal disk storage system, and reconnecting the transferred disk storage system to the failed server when it again becomes available. This minimizes the processing time required to make the disk storage contents identical, both at the time of failure and at the time of restoration.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AT	Austria	**GB**	United Kingdom	**MR**	Mauritania
AU	Australia	**GE**	Georgia	**MW**	Malawi
BB	Barbados	**GN**	Guinea	**NE**	Niger
BE	Belgium	**GR**	Greece	**NL**	Netherlands
BF	Burkina Faso	**HU**	Hungary	**NO**	Norway
BG	Bulgaria	**IE**	Ireland	**NZ**	New Zealand
BJ	Benin	**IT**	Italy	**PL**	Poland
BR	Brazil	**JP**	Japan	**PT**	Portugal
BY	Belarus	**KE**	Kenya	**RO**	Romania
CA	Canada	**KG**	Kyrgystan	**RU**	Russian Federation
CF	Central African Republic	**KP**	Democratic People's Republic of Korea	**SD**	Sudan
CG	Congo			**SE**	Sweden
CH	Switzerland	**KR**	Republic of Korea	**SI**	Slovenia
CI	Côte d'Ivoire	**KZ**	Kazakhstan	**SK**	Slovakia
CM	Cameroon	**LI**	Liechtenstein	**SN**	Senegal
CN	China	**LK**	Sri Lanka	**TD**	Chad
CS	Czechoslovakia	**LU**	Luxembourg	**TG**	Togo
CZ	Czech Republic	**LV**	Latvia	**TJ**	Tajikistan
DE	Germany	**MC**	Monaco	**TT**	Trinidad and Tobago
DK	Denmark	**MD**	Republic of Moldova	**UA**	Ukraine
ES	Spain	**MG**	Madagascar	**US**	United States of America
FI	Finland	**ML**	Mali	**UZ**	Uzbekistan
FR	France	**MN**	Mongolia	**VN**	Viet Nam
GA	Gabon				

METHOD FOR IMPROVING DISK MIRRORING ERROR RECOVERY IN A COMPUTER SYSTEM INCLUDING AN ALTERNATE COMMUNICATION PATH

SPECIFICATION

To all whom it may concern:

Be it known that Richard Rollins, Michael Ohran, Randall C. Johnson, Scott Bonsteel, and Richard S. Ohran, citizens of the United States of America, have invented a new and useful invention entitled METHOD FOR IMPROVING ERROR RECOVERY PERFORMANCE IN A FAULT-TOLERANT COMPUTER SYSTEM of which the following comprises a complete specification.

METHOD FOR IMPROVING ERROR RECOVERY PERFORMANCE IN A FAULT-TOLERANT COMPUTER SYSTEM

Microfiche Appendix. This specification includes a Microfiche Appendix which includes 1 page of microfiche and a total of 13 frames. The Microfiche Appendix includes computer source code illustrative of one preferred embodiment of the present invention.

Background of the Invention

Field of the Invention. This invention relates to fault-tolerant computer systems, and in particular to the methods used to recover from a computer failure in a system with redundant computers each with its own mass storage system(s).

Description of Related Art. It is often desirable to provide continuous operation of computer systems, particularly file servers which support a number of user workstations or personal computers on a network. To achieve this continuous

operation, it is necessary for the computer system to be tolerant of software and hardware problems or faults. This is generally done by having redundant computers and redundant mass storage systems, such that a backup computer or disk drive is immediately available to take over in the event of a fault.

A number of techniques for implementing a fault-tolerant computer system are described in Major et al., United States Patent 5,157,663, which is hereby incorporated by reference in its entirety, and Major's cited references. In particular, the invention of Major provides a replicated network file server capable of recovering from the failure of either the computer or the mass storage system of one of the two file servers. It has been used by Novell to implement its SFT-III fault-tolerant file server product.

Figure 1 illustrates the hardware configuration for a fault-tolerant computer system 100, such as described in Major. There are two server computer systems 110 and 120 connected to

network 101, from which they receive requests from client computers. While we refer to computers 110 and 120 as "server computer systems" or simply "servers" and show them in that role in the examples herein, this should not be regarded as limiting the present invention to computers used only as servers for other computer systems.

Server computer system 110 has computer 111 which includes a central processing unit and appropriate memory systems and other peripherals. Server computer system 120 has computer 121 which includes a central processing unit and appropriate memory systems and other peripherals. Mass storage systems 112 and 113 are connected to computer 111, and mass storage systems 122 and 123 are connected to computer 121. Mass storage systems 112 and 123 are optional devices for storing operating system routines and other data not associated with read and write requests received from network 101. Finally, there is an optional communications link 131 between computers 111 and 121.

The mass storage systems can be implemented using magnetic disk drives, optical discs, magnetic tape drives, or any other medium capable of handling the read and write requests of the particular computer system.

An operating system or other control program runs on server computer systems 110 and 120, executed by computers 111 and 121, respectively. This operating system handles server requests received from network 101 and controls mass storage systems 112 and 113 on server 110, and mass storage systems 122 and 123 on server 120, as well as any other peripherals attached to computers 111 and 121.

While Figure 1 illustrates only two server computer systems 110 and 120, because that is the most common (and lowest cost) configuration for a fault-tolerant computer system 100, configurations with more than two server computer systems are possible and do not depart from the spirit and scope of the present invention.

In normal operation, both server computer system 110 and server computer system 120 handle each mass storage write request received from network 101. Server computer system 110 writes the data from the network request to mass storage system 113, and server computer system 120 writes the data from the network request to mass storage system 122. This results in the data on mass storage system 122 being the mirror image of the data on mass storage system 113 and the states of server computer systems 110 and 120 are generally consistent. In the following discussion, the process of maintaining two or more identical copies of information on separate mass storage systems is referred to as "mirroring the information".

(For read operations, either server computer system 110 or server computer system 120 can handle the request without involving the other server, since a read operation does not change the state of the information stored on the mass storage systems.)

Although computer system 100 provides a substantial degree of fault tolerance, when one of server computer systems 110 or 120 fails, the fault tolerance of the system is reduced. In the most common case of two server computer systems, as illustrated by Figure 1, the failure of one server computer system results in a system with no further tolerance to hardware faults or many software faults.

In a fault-tolerant computer system such as described above, it is necessary after a failed server computer system has been restored to bring the previously-failed computer system into a state consistent with the server computer system that has continued operating. This requires writing all the changes made to the mass storage system of the non-failing server to the mass storage system of the previously-failed server so that the mass storage systems again mirror each other. Until that has been accomplished, the system is not fault tolerant even though the failed server has been restored.

If a server has been unavailable due to its failure for a period of time during which there have been only a limited number of changes made to the mass storage system of the non-failing server, it is possible for the non-failing server to remember all the changes made (for example, by keeping them in a list stored in its memory) and forward the changes to the previously-failed server when it has been restored to operation. The previously-failed server can then update its mass storage system with the changes and make it consistent with the non-failing server. This process typically does not cause excessive performance degradation to the non-failing server for any substantial period of time.

However, if there have been more changes than can be conveniently remembered by the non-failing server, then the non-failing server must transfer all the information from its mass storage system to the previously-failed server for writing on its mass storage system in order to ensure that

the two servers are consistent. This is a very time consuming and resource-intensive operation, especially if the non-failing server must also handle server requests from the network while this transfer is taking place. For very large mass storage systems, as would be found on servers commonly in use today, and with a reasonably high network request load, it might be many hours before the mass storage systems are again consistent and the system is again fault tolerant. Additionally, the resource-intensiveness of the recovery operation can cause very substantial performance degradation of the non-failed server in processing network requests.

Summary of the Invention

It is an object of the present invention to provide tolerance to disk faults even though the computer of a server computer system has failed. This is achieved by electronically switching the mass storage system used for network requests from the failed server computer system to the non-

failing server computer system. After the mass storage system from the failed server computer system has been connected to the non-failing server's computer, it is made consistent with the mass storage system of the non-failing server. This is typically a quick and simple operation. From that point on, the mass storage system from the failed server it is operated as a mirrored disk system, with each change being written by the non-failing server's computer to both the non-failing server's original mass storage system and to the mass storage system previously on the failed server and now connected to the non-failing server's computer.

While operating in this mode, the system will no longer be tolerant to processor failures if the non-failing server is the only remaining server (as would be the case in the common two-server configuration described above), but the system would be tolerant to failures of one of the mass storage systems.

It is a further object of the present invention to minimize the time the system is not fault tolerant by eliminating the need for time-consuming copying of the information stored on the mass storage system of the non-failing server to the mass storage of the previously-failed server to make the two mass storage systems again consistent and permit mirroring of information again.

This is also achieved by electronically switching the mass storage system from the failed server computer system to the non-failing server computer system. If this switch is accomplished after there have been only a small number of changes to the mass storage system of the non-failing server, the mass storage system from the failed server computer system can be quickly updated and made consistent, allowing mirroring to resume.

Furthermore, since the mirroring of the invention keeps the information on the mass storage system from the failed server consistent while it

is connected to the non-failing sever computer system, when the mass storage system is reconnected to the previously-failed server only those changes made between the time it was disconnected from the non-failed server and when it becomes available on the previously-failed server need to be made before it is again completely consistent and mirroring by the two servers (and full fault tolerance) resumes. This results in avoiding the substantial performance degradation experienced by the non-failing server during recovery using the prior art recovery method described above. As a result, the invention provides rapid recovery from a fault in the system.

These and other features of the invention will be more readily understood upon consideration of the attached drawings and of the following detailed description of those drawings and the presently preferred embodiments of the invention.

Brief Description of the Drawings

Figure 1 illustrates a prior art implementation of a fault-tolerant computer system with two server computer systems.

Figure 2 illustrates the fault-tolerant computer system of Figure 1, modified to permit the method of the invention by including means for connecting a mass storage system to either server's computer.

Figure 3 is a flow diagram illustrating the steps to be taken when a processor failure is detected.

Figure 4 is a flow diagram illustrating the steps to be taken when the previously-failed processor becomes available.

Detailed Description of the Invention

Referring to fault-tolerant computer system 200 of Figure 2, and comparing it to prior art fault-tolerant computer system 100 as illustrated in Figure 1, we see that mass storage systems 113 and 122, which were used for storing the information read or written in response to

requests from other computer systems on network 101, are now part of reconfigurable mass storage system 240. In particular, mass storage system 113 can be selectively connected by connection means 241 to either computer 111 or computer 121 (or possibly both computers 111 and 121, although such dual connection is not necessary for the present invention), and mass storage system 122 can likewise be independently selectively connected to either computer 111 or computer 121 by connection means 241. The mass storage system 240 is reconfigurable because of the ability to select and change connections between mass storage devices and computers.

While Figure 2 illustrates the most common dual server configuration anticipated by the inventors, other configurations with more than two servers are within the scope of the present invention, and the extension of the techniques described below to other configurations will be obvious to one skilled in the art.

There are a number of ways such connection means 241 can be implemented, depending on the nature of the mass storage system interface to computers 111 or 121. Connection means 241 can be two independent two-channel switches, which electronically connect all the interface signals from a mass storage system to two computers. Such two-channel switches may be a part of the mass storage system (as is common for mass storage systems intended for use with mainframe computers) or can be a separate unit. A disadvantage of using two-channel switches is the large number of switching gates that are necessary if the number of data and control lines in the mass storage interface is large. That number increases rapidly when there are more than two server computer systems in fault-tolerant computer system 200. For example, a fault-tolerant computer system with three computers connected to three mass storage systems would require 2.25 times the number of switching gates as the system illustrated in Figure

2. (The number of switching gates is proportional to the number of computers times the number of mass storage systems.) The number of switching gates can be reduced by not connecting every mass storage system to every computer, although such a configuration would be less flexible in its reconfiguration ability.

Another implementation of connection means 241 is for both computer 111 and computer 121 to have interfaces to a common bus to which mass storage systems 113 and 122 are also connected. An example of such a bus is the small computer system interface (SCSI) as used on many workstations and personal computers. When a computer wishes to access a mass storage system, the computer requests ownership of the bus through an appropriate bus arbitration procedure, and when ownership is granted, the computer performs the desired mass storage operation. A disadvantage of this implementation is that only one computer (the one

with current bus ownership) can access a mass storage system at a time.

If it is desirable to use a standard SCSI bus as means 241 for connecting mass storage systems 113 and 122 to computers 111 and 121, and to allow simultaneous access of the mass storage systems 113 and 122 by their respective server's computers, computers 111 and 121 can each have two SCSI interfaces, one connected to mass storage system 113 and one connected to mass storage system 122. Mass storage system 113 will be on a SCSI bus connected to both computers 111 and 121, and mass storage system 122 will be on a second SCSI bus, also connected to both computers 111 and 121. If computer 111 or computer 121 is not using a particular mass storage system, it will configure its SCSI interface to be inactive on that mass storage systems particular bus.

In the preferred embodiment, a high-speed serial network between computers 111 and 121 and mass storage systems 113 and 122 forms connection

means 241. Each computer 111 contains an interface to the network, and requests to a mass storage system 113 or 122 are routed to the appropriate network interface serving the particular mass storage system. Although a bus-type network, such as an Ethernet, could be used, the network of the preferred embodiment has network nodes at each computer and at each mass storage system. Each node can be connected to up to four other network nodes. A message is routed by each network node to a next network node closer to the message's final destination.

For the fault-tolerant computer system configuration of Figure 2, one network connection from the node at computer 111 is connected to the node for mass storage system 113, and another network connection from the node at computer 111 is connected to the node for mass storage system 122. Similar connections are used for computer 121. Mass storage system 113's node is connected directly to the nodes for computers 111 and 121,

and mass storage system 122's node is similarly connected (but with different links) to computers 111 and 121. Routing of messages is trivial, since there is only one link between each computer and each mass storage system.

The particular connecting means 241 used to connect computers 111 and 121 to mass storage systems 113 and 122 is not critical to the method of the present invention, so long as it provides for the rapid switching of a mass storage system from one computer to another without affecting the operation of the computers. Any such means for connecting a mass storage system to two or more computers is usable by the method of the present invention.

The method of the present invention is divided into two portions, a first portion for reacting to a processor failure and a second portion for recovering from a processor failure. The first portion of the method of the present invention is illustrated by Figure 3, which is a

flow diagram illustrating the steps to be taken when a processor failure is detected. The description of the method provided below should be read in light of Figure 2. For purposes of illustration, it will be assumed that connection means 241 initially connects mass storage system 113 to computer 111 and mass storage system 122 to computer 121, providing an equivalent to the configuration illustrated in Figure 1 although the connection means 241 of Figure 2 facilitates this equivalent configuration. Information mirroring as described above is being performed by computers 111 and 122. It is also assumed that computer 121 has experienced a fault, causing server computer system 120 to fail.

The method starts in step 301, with each computer 111 and 122 waiting to detect a failure of another server's computer 111 and 122. Such failure can be detected by probing the status of the other server's computer by a means appropriate to the particular operating system being used and

the communications methods between the servers. In the case of Novell's SFT-III, the method will be running as a NetWare Loadable Module, or NLM, and be capable of communicating directly with the operating system by means of requests. The NLM will make a null request to the SFT-III process. This null request will be such that it will never normally run to completion, but will remain in the SFT-III process queue. (It will require minimal resources while it remains in the process queue.) In the event of a failure of server computer system 121, SFT-III running on server computer system 111 will indicate the failure of the null request to the NLM of the method, indicating the failure of server 121. Because a processor failure has been detected, the method depicted in Figure 3 proceeds to step 302.

In step 302, detection of the failure of server 121 causes the discontinuation of mirroring information on the failed server 121. This discontinuation can either be done automatically by

the operating system upon its detection of the failure of server 121, or by the particular implementation of the preferred embodiment of the method of the present invention. In the case of SFT-III, the discontinuation of mirroring on server 121 is performed by the SFT-III operating system. Step 303 of the method is performed next.

In step 303, SFT-III remembers all data not mirrored on server 121 following its failure as long as the amount of data to be remembered does not exceed the capacity of the system resource remembering the data. If the particular operating system does not remember non-mirrored data, step 303 would have to be performed by the particular implementation of the method of the present invention. The step of remembering all non-mirrored data could be performed by any technique known to persons skilled in the art.

Next, step 304 of the method sets connection means 241 to disconnect mass storage system 122 from computer 121 of failed server 120,

and to connect it to computer 111 of non-failing server 110. At this point, the method can quickly test mass storage system 122 to determine if it is the cause of the failure of server 120. If it is, there is no fault-tolerance recovery possible using the method, and mass storage system 122 can be disconnected from computer 111 at connection means 241. If mass storage system 122 is not the cause of server 120's failure, then the cause must be computer 121, and the method can continue to achieve limited fault tolerance in the presence of the computer 121's failure.

Step 305 commands the operating system of server 110 to scan for new mass storage systems, causing the operating system to determine that mass storage system 122 is now connected to computer 111, along with mass storage system 113. SFT-III will detect through information on mass storage systems 113 and 122 that they contain similar information, but that mass storage system 122 is not consistent with mass storage system 113. In

step 306, SFT-III will update mass storage system 122 using the information remembered at step 303 and, after the two mass storage systems are consistent (i.e., contain identical mirrored copies of the stored information), step 307 will begin mirroring all information on both mass storage systems 113 and 122 and resume normal operation of the system. If an operating system different than SFT-III does not provide this automatic update for consistency and mirroring, the implementation of the method will have to provide an equivalent service.

Note that when SFT-III is used, the only steps of the method that must be performed by the NETWARE loadable module are: (1) detecting the failure of server 120 (step 301), (2) setting communications means 241 to disconnect mass storage system 122 from computer 121 and connecting it to computer 111 (step 304), (3) determining if mass storage system 122 was the cause of the failure of server 120 (also part of step (304), and (4)

commanding SFT-III to scan for mass storage systems so that it finds the newly-connected mass storage system 122 (step 305). All the other steps are performed as part of the standard facilities of SFT-III. In other embodiments of the invention, responsibility for performing the steps of the method may be allocated differently.

Figure 4 is a flow diagram illustrating the second portion of the invention - the steps to be taken when previously-failed server 120 becomes available again. Server 120 would typically become available after correction of the problem that caused its failure described above. Step 401 determines that server 102 is available and the method proceeds to step 402. In step 402, the method sets connection means 241 to disconnect mass storage system 122 from computer 111 after commanding SFT-III on server 110 to remove mass storage system 122 from its active mass storage systems. Due to the unavailability of mass storage system 122 on server 110, data mirroring on server

110 will be stopped by SFT-III and it will begin remembering changes to mass storage system 113 not made to mass storage system 122 to be used in making the storage systems consistent later.

In step 403, mass storage system 122 is reconnected to computer 121, and in step 404, SFT-III on server 120 is commanded to scan for the newly-connected mass storage system 122. This returns mass storage system 122 to the computer 121 to which it was originally connected prior to a server failure. When SFT-III on server 120 detects mass storage system 122, it communicates with server 110 over link 131. At this point, the operating systems on servers 110 and 120 work together to make mass storage system 122 again consistent with mass storage system 113 (i.e., by remembering interim changes to mass storage system 113 and writing them to mass storage system 122), and when consistency is achieved, data mirroring on the two servers resumes. At this point, recovery from the server failure is complete.

In an SFT-III system, the only steps of the method that the NetWare Loadable Module must perform are: (1) detecting the availability of server 120 (step 401), (2) removing mass storage system 122 from the operating system on server 110 (step 402), (3) disconnecting mass storage system 122 from computer 111 and connecting it to computer 121 by setting connection means 241 (step 403), and (4) commanding SFT-III on server 120 to scan for mass storage so that it locates mass storage system 122 (step 404). The steps involved with making mass storage systems 113 and 122 consistent and reestablishing data mirroring (step 405) are performed as part of the standard facilities of SFT-III. In other embodiments of the invention, responsibility for the steps of the method may be allocated differently.

Figure 2 illustrates optional mass storage systems 112 and 123 attached to computers 111 and 121, respectively. While these two mass storage systems are not required by the method of

the present invention, they are useful during the restoration of a failed server. They provide storage for the operating system and other information needed by failed server 120 to begin operation before mass storage system 122 is switched from computer 111 to computer 121. Were mass storage system 123 not available, some means of having mass storage system 122 connected both to computer 121 (for initializing its operation following correction of its failure) and computer 111 (for continued disk mirroring) would be necessary. Alternatively, if the initialization time of server 120 is short, mass storage system 122 could be switched from computer 111 to computer 121 at the start of server 120's initialization, though this would result in more changes that must be remembered and made before data mirroring can begin again.

It is to be understood that the above described embodiments are merely illustrative of numerous and varied other embodiments which may

constitute applications of the principles of the invention. Such other embodiments may be readily devised by those skilled in the art without departing from the spirit or scope of this invention and it is our intent they be deemed within the scope of our invention.

Claims

We claim:

1. A method for rapid failure recovery and system restoration in a fault-tolerant computer system, said computer system comprising:

(A) a first server computer system, comprising a first computer executing an operating system;

(B) a second server computer system, comprising a second computer executing an operating system;

(C) a first mass storage system connected to said first computer;

(D) a second mass storage system; and

(E) means for connecting said second mass storage system to said first computer and to said second computer;

WHEREIN whenever said first computer writes data to said first mass storage system, said second

computer writes a mirror copy of said data to said second mass storage system,

the method comprising the steps of:

(1) detecting a failure of said second computer;

(2) discontinuing causing said writing of said mirror copy on said second mass storage system;

(3) remembering data written to said first mass storage system but not written to said second mass storage system;

(4) configuring said second mass storage system to record information from said first computer;

(5) writing said remembered data to said second mass storage system;

(6) whenever new data is written to said first mass storage system, writing a mirror copy of said new data to said second mass storage system;

(7) detecting said second computer's availability;

(8) reconfiguring said second mass storage system to record information from said second computer;

(9) reestablishing data mirroring such that whenever said first computer writes data to said first mass storage system, said second computer writes a mirror copy of said data on said second mass storage system.

2. A method as in claim 1 wherein step (1) is performed by said first computer.

3. A method as in claim 2 wherein step (2) is performed by said first computer.

4. A method as in claim 1 wherein step (3) is performed by said first computer.

5. A method as in claim 4 wherein step (5) is performed by said first computer.

6. A method as in claim 5 wherein step (6) is performed by said first computer.

7. A method as in claim 1, wherein said first mass storage system and said second mass storage system each comprise at least one magnetic disk drive.

8. A method as in claim 1, wherein said means for connecting said second mass storage system comprises a serial network.

9. A method as in claim 1 wherein said operating systems are the SFT-III operating system.

10. A method as in claim 9 wherein steps (1), (4) and (5) are performed by a NETWARE loadable module.

11. A method for rapid failure recovery and system restoration in a fault-tolerant computer system, said computer system comprising:

(A) a first server computer system, comprising a first computer executing an operating system;

(B) a second server computer system, comprising a second computer executing an operating system;

(C) a first mass storage system connected to said first computer;

(D) a second mass storage system; and

(E) means for selectively connecting said second mass storage system to said first computer and to said second computer;

WHEREIN in the absence of a fault said second mass storage system is connected to said second computer; and

WHEREIN whenever said first computer writes data to said first mass storage system said first computer can also cause said second computer to write a mirror copy of said data to said second mass storage system,

the method of the invention comprising:

(1) on said first computer, detecting a failure of said second computer;

(2) on said first computer, discontinuing causing said writing of said mirror copy on said second mass storage system by said second computer;

(3) on said first computer, remembering data written to said first mass storage system but not written to said second mass storage system;

(4) on said first computer, setting said means for connecting said second mass storage system to connect said second mass storage system to said first computer;

(5) on said first computer, commanding said operating system of said first computer to scan for mass storage systems such that said operating system of said first computer will determine that both said first mass storage system and said second mass storage system are now connected to said first computer;

(6) on said first computer, writing said remembered data to said second mass storage system;

(7) on said first computer, whenever new data is written to said first mass storage system,

writing a mirror copy of said new data to said second mass storage system;

(8) on said first computer, detecting said second computer's availability;

(9) on said first computer, commanding said operating system of said first computer to remove said second mass storage system;

(10) setting said means for connecting said second mass storage system to connect said second mass storage system to said second computer;

(11) on said second computer, commanding said operating system of said second computer to scan for mass storage systems such that said operating system of said second computer will determine that said second mass storage system is now connected to said second computer;

(12) reestablishing data mirroring such that whenever said first computer writes data to said first mass storage system said first

computer also causes said second computer to write a mirror copy of said data on said second mass storage system.

12. A method as in claim 11, wherein said first mass storage system and said second mass storage system each comprise at least one magnetic disk drive.

13. A method as in claim 12, wherein said means for connecting said second mass storage system comprises a serial network.

14. A method for rapid failure recovery in a fault-tolerant computer system, said computer system comprising:

(A) a first server computer system, comprising a first computer executing an operating system;

(B) a second server computer system, comprising a second computer;

(C) a first mass storage system connected to said first computer;

(D) a second mass storage system; and

(E) means for selectively connecting said second mass storage system to said first computer and to said second computer;

WHEREIN in the absence of a fault said second mass storage system is connected to said second computer; and

WHEREIN whenever said first computer writes data to said first mass storage system said first computer can also cause said second computer to write a mirror copy of said data on said second mass storage system,

the method of the invention comprising said first computer performing the steps of:

(1) detecting a failure of said second computer;

(2) discontinuing causing said writing of said mirror copy on said second mass storage system by said second computer;

(3) remembering data written to said first mass storage system but not written to said second mass storage system;

(4) setting said means for connecting said second mass storage system to connect said second mass storage system to said first computer;

(5) commanding said operating system of said first computer to scan for mass storage systems such that said operating system of said first computer will determine that both said first mass storage system and said second mass storage system are now connected to said first computer;

(6) writing said remembered data to said second mass storage system;

(7) whenever new data is written to said first mass storage system, writing a mirror copy of said new data to said second mass storage system.

15. A method as in claim 14, wherein said first mass storage system and said second mass storage system each comprise at least one magnetic disk drive.

16. A method as in claim 15, wherein said means for connecting said second mass storage system comprises a serial network.

17. A method for system restoration in a fault-tolerant computer system, said computer system comprising:

(A) a first server computer system, comprising a first computer executing an operating system;

(B) a second server computer system, comprising a second computer executing an operating system;

(C) a first mass storage system connected to said first computer;

(D) a second mass storage system; and

(E) means for connecting said second mass storage system to said first computer and to said second computer;

WHEREIN said second computer is initially unavailable for use, and

WHEREIN said second mass storage system is initially connected to said first computer, the method comprising:

(1) on said first computer, detecting said second computer's availability;

(2) on said first computer, commanding said operating system of said first computer to remove said second mass storage system;

(3) setting said means for connecting said second mass storage system to connect said second mass storage system to said second computer;

(4) on said second computer, commanding said operating system of said second computer to scan for mass storage systems such that said operating system of said second computer will

determine that said second mass storage system is now connected to said second computer;

(5) reestablishing data mirroring such that whenever said first computer writes data to said first mass storage system said first computer also causes said second computer to write a mirror copy of said data on said second mass storage system.

18. A method as in claim 17, wherein said first mass storage system and said second mass storage system each comprise at least one magnetic disk drive.

19. A method as in claim 18, wherein said means for connecting said second mass storage system comprises a serial network.

20. A method as in claim 17 wherein said operating system is the SFT-III operating system.

21. A method as in claim 20 wherein steps (1), (4) and (5) are performed by a NETWARE loadable module.

22. A method for rapid failure recovery in a fault-tolerant computer system, said computer system comprising:

(A) a first server computer system, comprising a first computer executing an operating system;

(B) a second server computer system, comprising a second computer executing an operating system;

(C) a first mass storage system connected to said first computer;

(D) a second mass storage system; and

(E) means for connecting said second mass storage system to said first computer and to said second computer;

WHEREIN whenever said first computer writes data to said first mass storage system, said second computer writes a mirror copy of said data to said second mass storage system,

the method comprising the steps of:

(1) detecting a failure of said second computer;

(2) discontinuing causing said writing of said mirror copy on said second mass storage system;

(3) remembering data written to said first mass storage system but not written to said second mass storage system;

(4) configuring said second mass storage system to record information from said first computer;

(5) writing said remembered data to said second mass storage system; and

(6) whenever new data is written to said first mass storage system, writing a mirror copy of said new data to said second mass storage system.

23. A method for system restoration in a fault-tolerant computer system, said computer system comprising:

(A) a first server computer system, comprising a first computer executing an operating system;

(B) a second server computer system, comprising a second computer executing an operating system;

(C) a first mass storage system connected to said first computer;

(D) a second mass storage system;

(E) means for connecting said second mass storage system to said first computer and to said second computer;

WHEREIN said second computer is initially unavailable for use; and

WHEREIN said second mass storage system is initially configured to record information from said first computer,

the method comprising the steps of:

(1) detecting said second computer's availability;

(2) reconfiguring said second mass storage system to record information from said second computer;

(3) establishing data mirroring such that whenever said first computer writes data to said first mass storage system, said second computer writes a mirror copy of said data on said second mass storage system.

24. A method for rapid failure recovery and system restoration in a fault-tolerant computer system, the method comprising the steps of:

(1) obtaining a computer system, the computer system comprising:

(A) a first server computer system, comprising a first computer executing an operating system;

(B) a second server computer system, comprising a second computer executing an operating system;

(C) a first mass storage system connected to said first computer;

(D) a second mass storage system; and

(E) means for connecting said second mass storage system to said first computer and to said second computer;

(2) operating said computer system such that absent a fault, whenever said first computer writes data to said first mass storage system, said second computer writes a mirror copy of said data to said second mass storage system;

(3) detecting a failure of said second computer;

(4) discontinuing causing said writing of said mirror copy on said second mass storage system;

(5) remembering data written to said first mass storage system but not written to said second mass storage system;

(6) configuring said second mass storage system to record information from said first computer;

(7) writing said remembered data to said second mass storage system;

(8) whenever new data is written to said first mass storage system, writing a mirror copy of said new data to said second mass storage system;

(9) detecting said second computer's availability;

(10) reconfiguring said second mass storage system to record information from said second computer;

(11) reestablishing data mirroring such that whenever said first computer writes data to said first mass storage system, said second computer writes a mirror copy of said data on said second mass storage system.

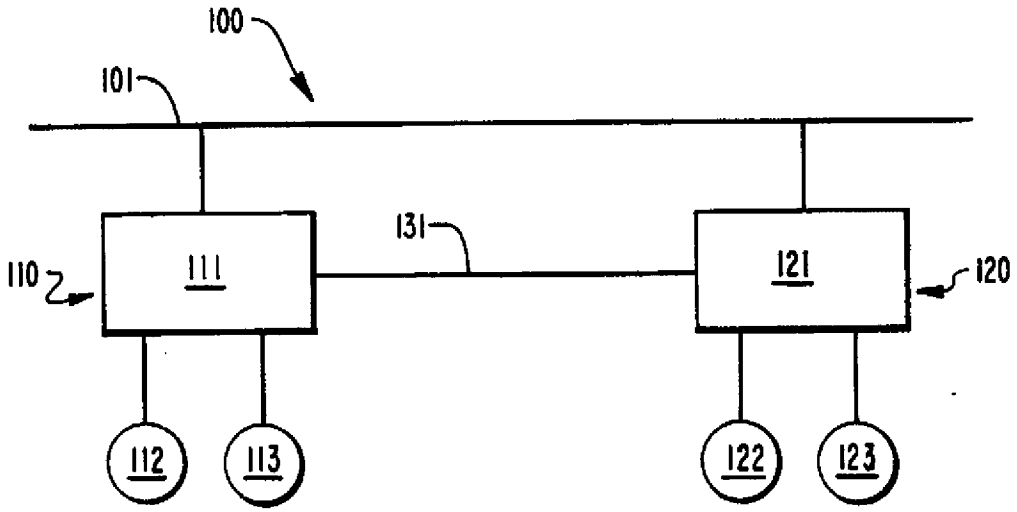


FIG. 1
(PRIOR ART)

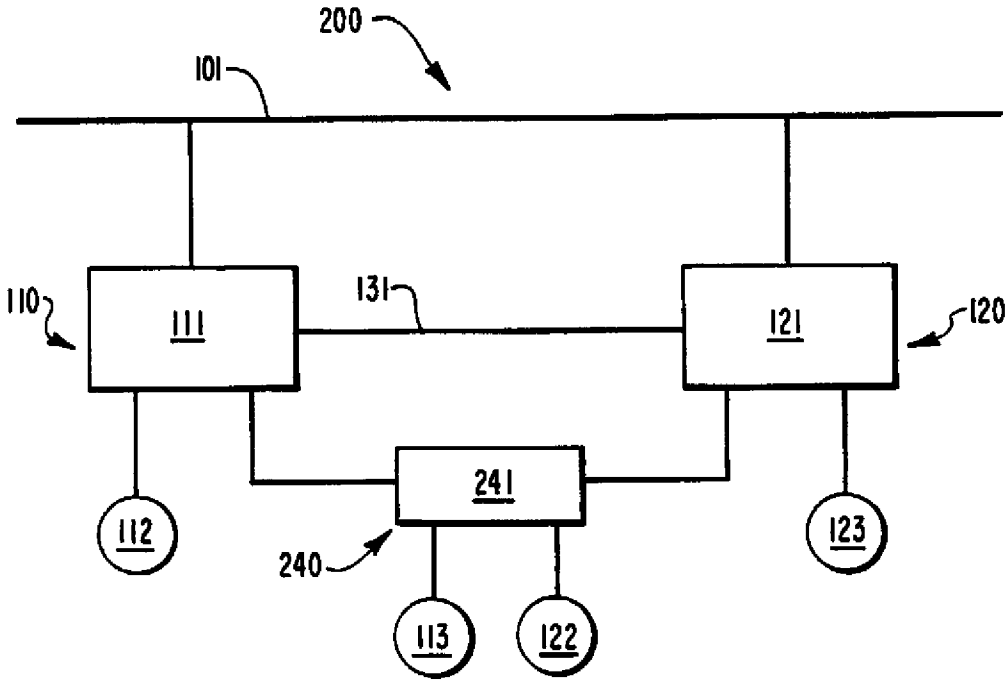


FIG. 2

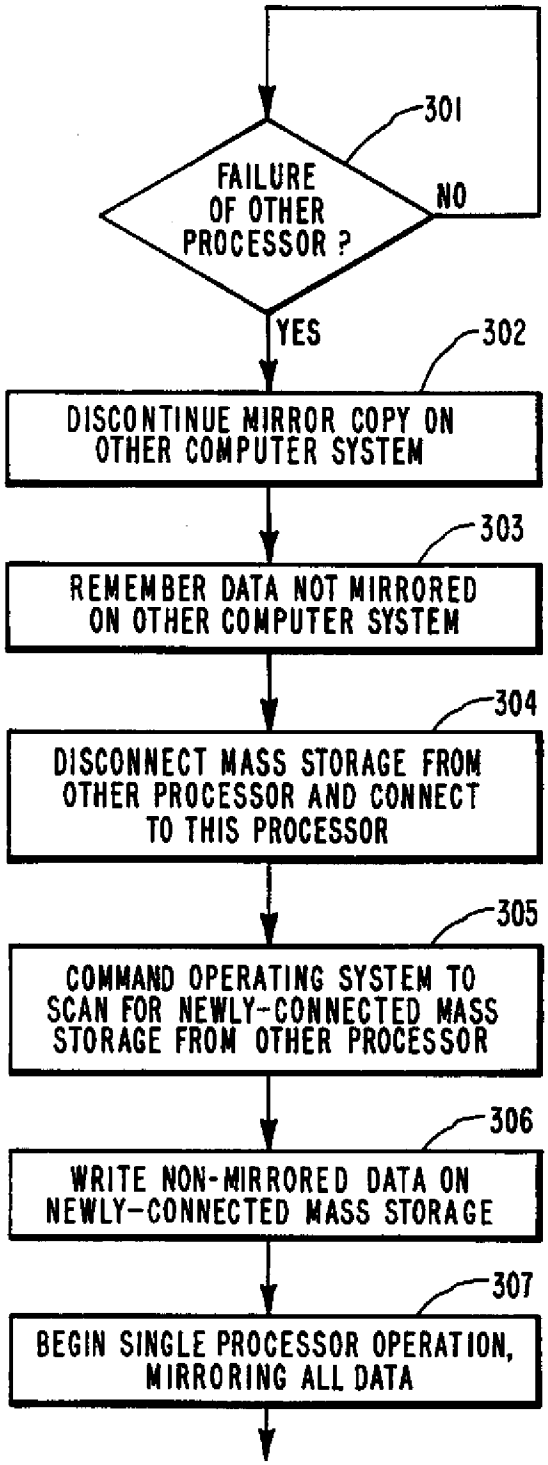

301
FAILURE OF OTHER PROCESSOR ?
NO
YES
302
DISCONTINUE MIRROR COPY ON OTHER COMPUTER SYSTEM
303
REMEMBER DATA NOT MIRRORED ON OTHER COMPUTER SYSTEM
304
DISCONNECT MASS STORAGE FROM OTHER PROCESSOR AND CONNECT TO THIS PROCESSOR
305
COMMAND OPERATING SYSTEM TO SCAN FOR NEWLY-CONNECTED MASS STORAGE FROM OTHER PROCESSOR
306
WRITE NON-MIRRORED DATA ON NEWLY-CONNECTED MASS STORAGE
307
BEGIN SINGLE PROCESSOR OPERATION, MIRRORING ALL DATA

FIG. 3

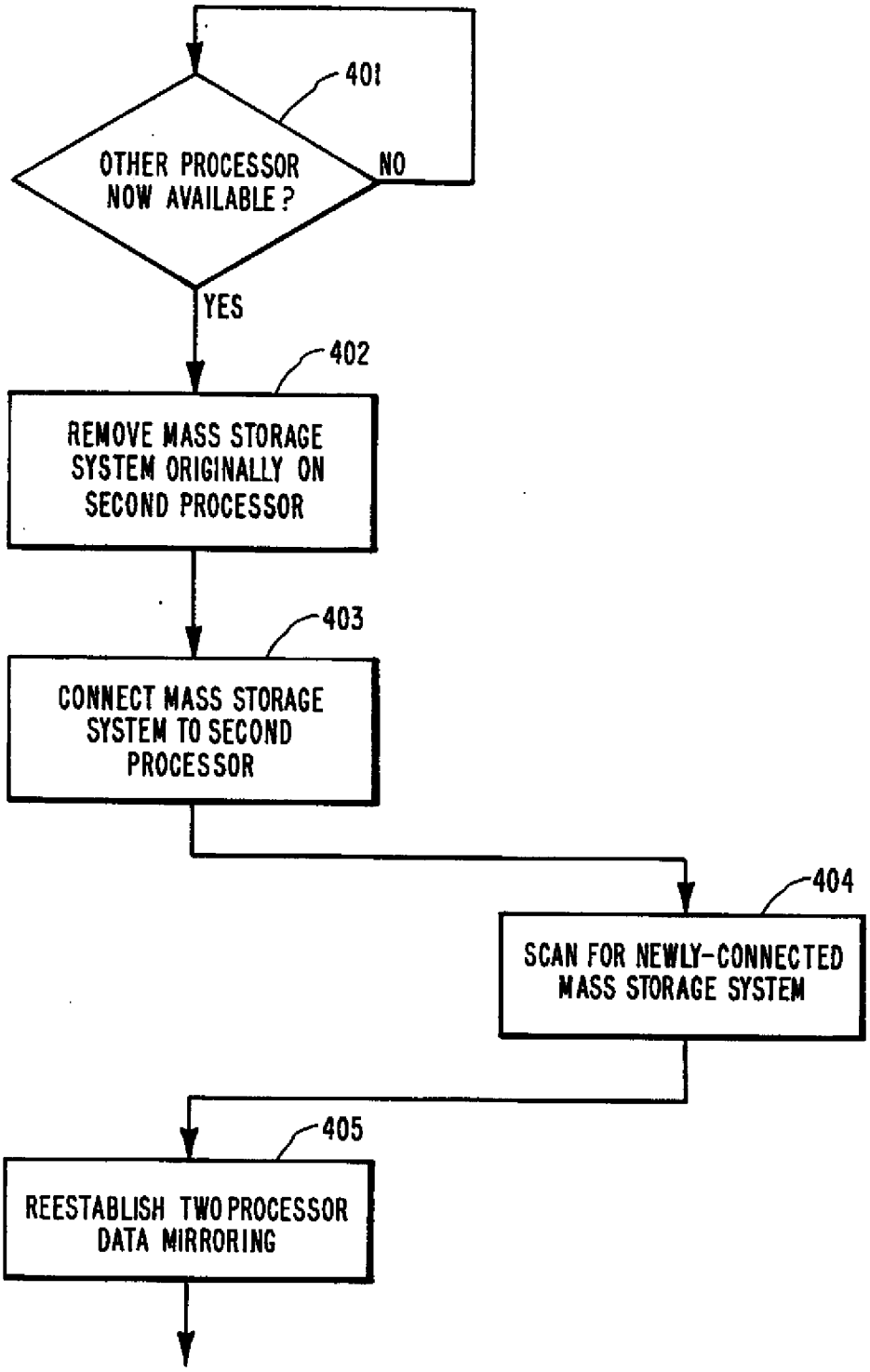

401
OTHER PROCESSOR NOW AVAILABLE?
NO
YES
402
REMOVE MASS STORAGE SYSTEM ORIGINALLY ON SECOND PROCESSOR
403
CONNECT MASS STORAGE SYSTEM TO SECOND PROCESSOR
404
SCAN FOR NEWLY-CONNECTED MASS STORAGE SYSTEM
405
REESTABLISH TWO PROCESSOR DATA MIRRORING

FIG. 4